



Microcontroller Based Programmable Logic Controller and Graphical User Interfacing Using Visual Basic

Hiren Chothani ¹ Bhavin Changela ² Kiritkumar Saravaiya ³

Instrumentation & Control Department^{1,3}

Biomedical Department²

A.V.Parekh Technical Institute, Rajkot^{1,2,3}

Abstract—Teaching and learning PLC in the laboratory has been traditionally carried out using branded simulators and supporting evaluation boards. The branded PLC are not widely used because their high cost and limited accessibility. This paper presents *EasyLadder*, an integrated development environment (IDE) tool developed for teaching and learning the PLC using Philip's controller. We used P89v51rd2 controller because it has a enough ports and ISP facility. As a beginner, instead of purchasing costly PLCs, it's good to make our own PLC with simple IDE. The system allows students to write their own ladder logic programs, simulate it and test on hardware.

Keywords—PLC; Visual Basic; Microcontroller; GUI; 'c' Language; Serial Communication; protocols; Ladder Logic; ISP.

I. INTRODUCTION

Programmable logic controller (PLC) is a microprocessor-based control system which is the most popular controller used in automation of industry manufacture. But in current market, PLCs of different companies have their own systems; especially have Different environments of programming and debugging.

It is used in process control for simple switching tasks, PID control, complex data manipulation, arithmetic operations, timing and process and machine control and coordination. As technology has advanced over the years, especially microprocessor technology, PLC capability has grown from primarily on/off control to include operator interfaces, performing arithmetic operations and data manipulation, computer communications, supplemental computer memory, analog and positioning control, and remote I/O. Today, PLC's are used in all facets of industry, provide a broad range of functions, can be programmed using a desktop personal computer, and can be purchased for as little as several hundred dollars. With such broad use in industry, wide-ranging application in engineering related fields and availability to users at all levels, it is imperative that engineering educators

provide young engineers with a fundamental understanding of the operation and capabilities of PLCs. So we have started our dissertation and research works on to building of a PLC platform. This platform is unique in that it is designed to allow students to self-learn fundamental PLC operation [7].

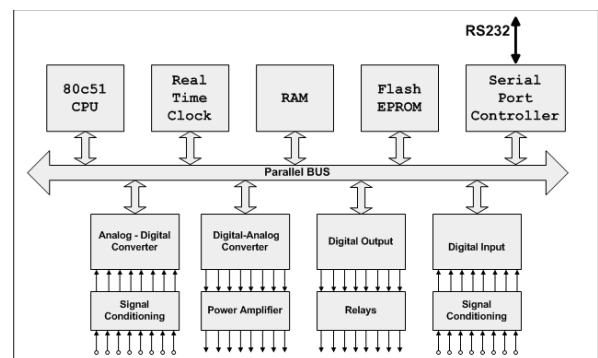


Figure 1. Structure of a PLC.

Also students can perform ladder logic programming, expose students on ladder logic experiments theoretically and practically and how to developed their own ladder logic editor and interfacing to the controller.

Software part include real time programming using Microsoft VB6 for developing IDE as well as 'c' programming for controlling different I/O using Philips controller.

GUI systems are becoming increasingly popular against traditional systems due to their ease of use. However, GUI systems are often challenging to design due to their complexity and some special features. But Visual Basic 6 gives us a big platform to overcome the complexity of designing complex GUI. We present a Ladder Logic GUI Editor using VB6. The GUI model consists of the hierarchical

planning of controls which representing the possible events in the GUI. We have developed GUI which is used for ladder logic simulator as well as editor with drag and drop aspects of GUI which is much challenging jobs for students with little or no visual acuity. Interactive software will be equipped with set of graphic instructions to ease any mistake when performing experiments. The software will be integrated with hardware to make a real GUI to experience a real programmable logic controller.

II. ARCHITECTURE OF PLC

A. Design of Hardware

The traditional of the hardware structure of PLC is composed of CPU and peripheral equipments (for example storage, A/D converter, D/A converter and etc) and I/O interface.

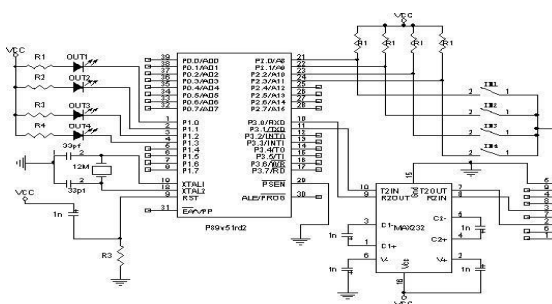
The structure of the PLC controller we have developed was simplified through using a powerful Philips controller. For the reason that the MCU is integrated with enough storage, such a new structure can A/D converter, D/A converter and etc, the structure of our PLC was simplified into being composed of CPU and interfaces. As a result, reduce the cost and improve the job dependability. The structure of general PLC is shown in Fig. 1. The hardware of our PLC including CPU, switch inputs, A/D input, relay controlling output, D/A output, RS232 interface, external clock, EPROM and etc.

The blockdiagram in Fig.2. Shows the hardware section of our testing PLC. The Push button is connected to the terminalP2 port with pull up resistors. The LEDs are connected to the terminalP1 port with current limiting resisters connected in series with the LED. In this case controller sinks the current from power supply so it will prevent the controller to be handling heavy current. The terminalTxD and RxD pins are connected to the MAX232 RxD and TxD pins [13]. The terminal TxD and RxD of Max232 also connected to Serial port of Computer using DB9 Connector. The crystal used for P89v51rd2 is 11.059MHz.

B. Design of Software

In the design of PLC system, it is very important to offer a good IDE for the users. So we have this work as a key point of our whole design work. Our IDE uses normal personal computer (PC) as the basic programming tool. In the point of programming language, our design is based on PLC Programming Language Standard.

Figure 2. PLC Testing Board



We choose the most popular languages: Ladder Diagram (LD) as the programming languages for users.

The purpose of designing an IDE is to offer an intuitional, convenient, efficient platform for programming of PLC. The PLC platform generally have following function: Editing, Compiling, simulating and downloading. For designing purpose entire GUI Can be divided into several independent modules, which are shown in Fig. 3.

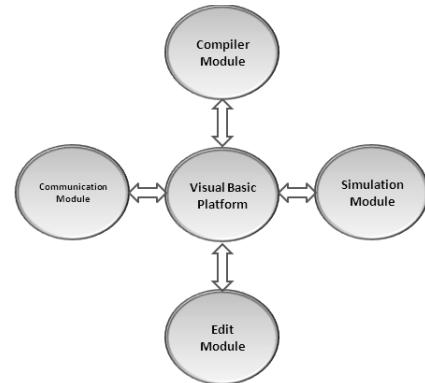


Figure 3. The components of PLC's Integrated Development environment

Edit Module: It is a basic module of function, which offers a programming environment for Ladder Logic. In this module user can edit the logic of the application using the controls on the GUI. User can change the input and output contacts of each rung by selecting each images on the GUI using Mouse button. During the simulation mode each control are disable except simulation stop command in Menu Editor. We can enable each control during editing by following way.

```

Cmd_NC.Enable = true
Cmd_NO.Enable = true
Cmd_COIL.Enable = true
Cmd_TIMER.Enable = true
Cmd_RUN.Enable = true
Cmd_COMPILE.Enable = true
Cmd_STOP.Enable = true
Cmd_LOAD.Enable = true
    
```

Where Cmd_NC, Cmd_NO, Cmd_COIL are command buttons. After editing user must compile the editing code to checking the syntax and proper labeling of the controls.

2) Compile Module: The compiler is used to compile the Ladder Diagram languages into the destination code. This module also taking care of the labeling of each controls used in ladder diagram. On error it will display the error message on screen using MessageBox function of VB6. Compilation is necessary before run the simulation. The Compilation code generated after the compilation is shown below.

```
#111111111111111111111111111111111111@0000000000000000
0000000000A1xxxx5xxxxxxxxxxxxxxxxxxxxxxBxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxC56yyyK0N0L0M0?
```

This code will be transfer to PLC using serial communication module. These codes generated as per the logic develop by programmer.

3) *Communication Module*: This module is in charge of the communication between the Personal computer and the PLC hardware. In one way, the Communication Module has the function of sending message to the PLC hardware, including downloading user's program and setting of the parameters of PLC hardware. In another way, the module also has the functions of reading a status of PLC elements and respective changes in running GUI.

The easiest way to access COM ports in Visual Basic 6 is by placing the Microsoft Comm. control on a form, and writing code to read and write each serial port. MSComm. Control come with Visual Basic 6 and also we will need one Microsoft Comm. control per one serial port. We can set the setting of communication port as shown below[1].

```
Private Sub Form_Load ()

MSComm1.CommPort = 1' comm. port no.
MSComm1.Settings = "2400, N, 8, 1"
MSComm1.DTREnable = False
MSComm1.PortOpen = True

End Sub
```

4) *Simulation Module*: The Simulation Module is in charge of the full continuous simulation of the users' program. The main objective of this module is to continuously reading the state of the input devices and updates the state of output devices as per the application of the program.

It's not easy to designing the infinite loop for simulation module. Because of the operating system which is install in PC might be loss their control over processor, so entire system become stop to working. For the solution of that we have use DoEvents () function in VB6 which temporarily gives control to the operating system so that it can process other events.

C. Design of Device driver

Before communicate between PLC and PC through GUI, it is necessary to load driver of PLC in a memory of controller. Without driver PC can't communicate to PLC or vice a versa. We have develop a driver using 'C' language with a Keil compiler for editing driver code for communication as well as configuration code for P89v51Rd2 controller. Wehave used flashmagic software to load driver into controller using serial

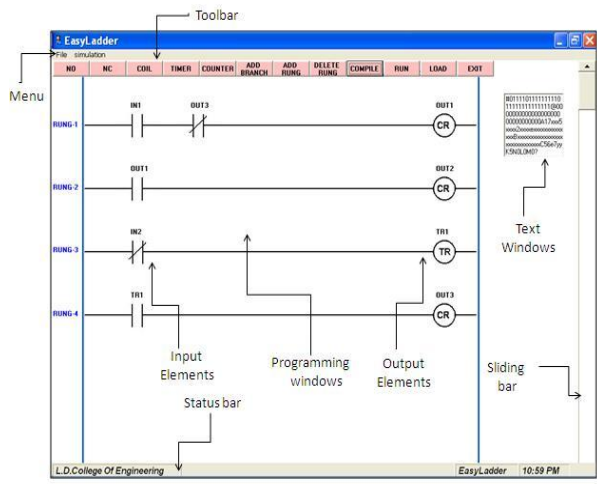


Figure. 4: GUI Screen for given Example

port. We don't need to purchase separate programmer to load driver due to in-system programming (ISP) facility in controller.

III. THE RESULT AND DISCUSSION

This section describes an intermediate level example to be developed by the students. It has four rungs with three output coil, two input switches, one timer and four NO/NC contacts as shown in Fig.4.

Step1: Edit Module: In editing mode all controls are enabling to editing the Ladder Diagram except RUN Command As shown in Fig.4. To enable the RUN Command, user must compile the code before simulating the LD. During Compilation Progress bar at the bottom of the windows indicates the status of compilation process.

Step2: Compilation Module: After successful completion of compilation process, The Message box appeared on the screen as shown in Fig.5. After successful compilation, RUN Button is enable to simulate the Ladder Logic. The simulation will start after click on the RUN button on GUI. During the simulation all controls are disabling except Simulation STOP command. Shown in Fig.6.

Step3: Run Module: In this mode user can select the RUN

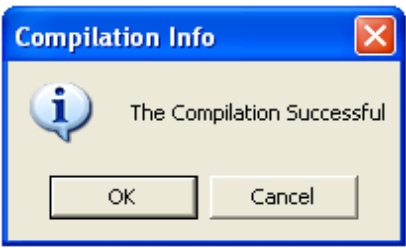


Figure 5. Message Box

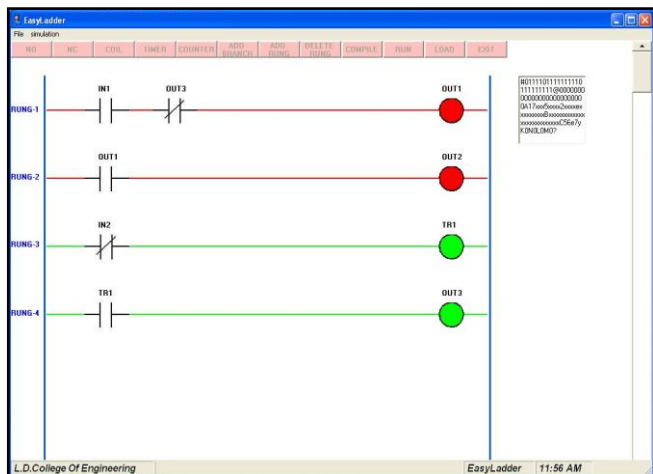


Figure 6. GUI with opcodes

button to simulating the LD. The status of each control on GUI after selecting the RUN button is shown in Fig.6.

The Result and status of I/O devices shown below.

Case: 1 (Fig.6)

IN1 = False
 IN2 = True
 OUT1 = De-Energized
 OUT2 = De-Energized
 TR1 = Energized
 OUT3 = Energized (AfterTR1 Seconds)

Case: 2 (Fig.7)

IN1 = True
 IN2 = False
 OUT1 = Energized
 OUT2 = Energized
 TR1 = De-Energized

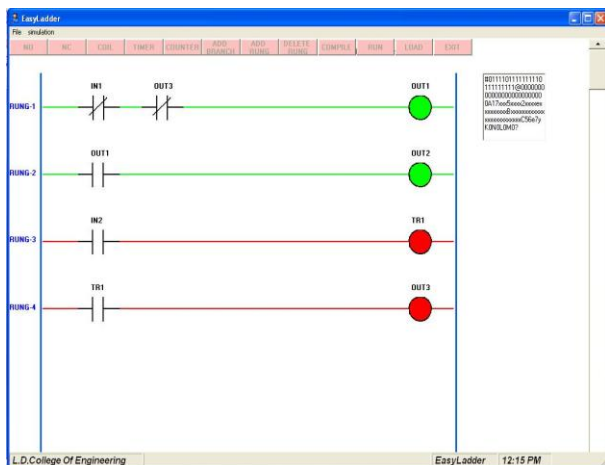


Figure 7. GUI with Running Status

OUT3 = De-Energized

IV. DISCUSSION

This entire platform is well built with the controls of VB6 and hardware part of PLC with identifying run smoothly. But as far as real time simulation is concern then it's very difficult to synchronize with real clock due to high speed of processor. But this problem can be solves if we use API function in vb6 to synchronize with speed of operating system.

V. CONCLUSION

A software tool and hardware for teaching/learning the microcontroller based PLC, called *EasyLadder*, and has been presented. It can be used in the laboratory when describing the PLC architecture, the Ladder logic instruction set or the embedded peripherals operation, and also to write and test Application programs. A copy of *EasyLadder* can be delivered to the students, so they can clear the fundamentals of the PLC before they enter the laboratory. The main advantages of this platform are its versatility, design flexibility and low cost compared to commercially available products as a teaching point of view. It is considered that this platform has an impact in the automation where PLC is the part of system and so student can understand heart of the system. The source codes of *EasyLadder* are open to student so that everyone makes their own small PLC using editing the codes. Also, the authors consider that this system is a platform on which a more powerful PLC can be developed in future. The results of the application of our approach to education have been excellent both in terms of students' motivation and knowledge.

ACKNOWLEDGMENT

I would like to express my deepest gratitude to staff members of Instrumentation and control and Biomedical department of A V Parekh Technical Institute, Rajkot, who contributed and inspired us to the overall success of this research work.

REFERENCES

- [1] Michael Vine, "Visual Basic Programming (For the absolute beginner)".
- [2] Vincet Himpe, "Visual Basic for electronics engineering applications".
- [3] David I. Schneider, "Computer Programming Concepts and Visual Basic".
- [4] Clayton Walnum, "Complete Idiot's Guide to Visual Basic 6" ISBN: 078971812x, Publication Date: 12/15/98.
- [5] Steven Holzner "Visual Basic 6 Black Book", ISBN:1576102831 Publication Date: 08/01/98.
- [6] Francesco Balena, "Programming Microsoft visual Basic 6". Copyrights (c) 1999 by Francesco Balena.
- [7] E.A. Parr, "Programmable Controllers An engineer's guide".
- [8] Hugh Jack, "Automating Manufacturing Systems with PLCs", Version 4.7, April 14, 2005.
- [9] L. A. Bryan, E. A. Bryan, "Programmable Controllers Theory and Implementation".
- [10] Kirk Zurell, "C Programming for Embedded Systems".
- [11] Ted Van Sickle, "Programming Microcontrollers in C Second Edition".
- [12] Peter van der Linden, "Expert C Programming: Deep C Secrets".
- [13] Dogan Ibrahim, "Microcontroller Projects in C for the 8051".